Mixed-Integer PDE-Constrained Optimal Control of Gas Networks*

Mirko Hahn†, Sven Leyffer‡, and Victor M. Zavala§

December 18, 2017

Abstract

We develop a mixed-integer optimal control model with partial differential equation (PDE) constraints for gas transport networks, designed for controlling extreme state transitions, such as flow reversals. Our model shows how to combine binary compressor controls with PDE flow models. We model the flow of gas using a variant of the Euler equations, which we discretize using a finite volume scheme, which obeys conservation of mass and average impulse independent of the direction of flow. The resulting large-scale mixed-integer nonlinear optimization problem is difficult to solve by using standard branch-and-bound solvers, and we propose a new tree-search strategy that allow us to solve realistic instances. We compare the performance of several solution schemes on twelve test instances, showing that our custom tree-search strategy greatly outperforms a generic strategy in most instances.

Keywords: Gas networks, flow reversal, PDE constraints, mixed-integer nonlinear optimization.

AMS-MSC2000: 35L60, 35L65, 90C06, 90C11, 90C30

1 Introduction: Background and Motivation

In this paper, we develop a mixed-integer optimal control model for gas transport networks that are not stationary. Instead, our model uses partial differential equation (PDE) constraints to model the flow of gas. Our goal is to find control configurations that satisfy given, deterministically fluctuating demand patterns at minimal network operation cost. We attempt to minimize the number of restrictions we place on control inputs. Most important, we avoid prior assumptions on fixed direction of flow such as those made in [15] and simplifications made concerning the operational constraints of compressor stations such as those found in [16]. This approach allows us to solve instances where the flow of gas must be reversed. In addition, in contrast to approximate solution methods as proposed by [4], we attempt to solve the resulting optimization problems to integer optimality and propose methods of accelerating this process.

Several efforts are underway to develop mathematical descriptions of natural gas transport networks and derive insights into their optimal operation. The range of prior work by other

*Preprint ANL/MCS-P7095-0817

†Mathematics and Computer Science Division, Argonne National Laboratory, Lemont, IL 60439, hahnm@anl.gov.

‡Mathematics and Computer Science Division, Argonne National Laboratory, Lemont, IL 60439, leyffer@anl.gov.

§Chemical and Biological Engineering, University of Wisconsin Madison, Madison, WI 53706, victor.zavala@wisc.edu.

researchers is too broad to properly address fully here and ranges from detailed descriptions of the operation of a single compressor station [5,10] to more ambitious work attempting to optimize controls for entire networks with integer switches that inhabit a steady state [9], approximate optimal transient controls on larger networks with integer switches [4], and even find optimal transient controls on simplified networks without integer switches with [15] and without [16] uncertainty.

Outline of paper. Section 2 introduces our model formulation, which is based on [15] and makes simplifying assumptions similar to those made in [4, 9, 16]. We avoid assumptions on fixed flow directions made in [15] by using a finite volume discretization scheme. In designing our model, we follow a modular approach with an eye to future efforts to increase model fidelity, scale the method to larger networks, and use parallel solvers. Section 3 proposes several modifications designed to lessen the performance impact of large numbers of integers in branch-and-bound solvers. Also discussed are possible extensions to increase the potential for future parallelization. Section 4 presents two test problems to which we apply our modified solvers to estimate the impact of our modifications. We find that performance can be increased substantially by modifying tree-search methods and that approximate solvers may benefit from branching rules that take into account the structure of the problem. Section 5 summarizes our findings and proposes future avenues of research.

Notation. We use uppercase calligraphic letters to denote sets. The lowercase letters i, j, k, l denote indices, the uppercase letters K, L denote break points in index sequences, and m, n, N denote set sizes. Vertices of the network graph are denoted by the letters v, w. Model quantities can be qualified by subscripts and superscripts. Parenthesized superscripts (e.g., $p^{(v)}$) denote the network element with which a quantity is associated. Noncursive subscripts (e.g., p_{in}) are generally symbolic and do not refer to variables. Quantities decorated with a horizontal bar (e.g., $\bar{p}$) are averages.

2 Optimization and Control Model

This section describes our gas network model, starting with the overall structure of the network graph in Section 2.1. Section 2.2 discusses the dynamics of gas networks and their discretization, while Section 2.3 describes the mechanism by which we control flow in our network. The complete optimization model is stated in Section 2.4.

2.1 Network Structure

Gas transport networks are complex systems that are commonly modeled using network components such as pipes, junctions, compressor machines, and various types of valves. We omit the modeling of valves and assume that valves and compressor machines occur only in larger assemblies called compressor stations. We assume such compressor stations to occur only at junctions, described in Section 2.3.

The structure of a network can be described by using a directed graph $G = (\mathcal{V}, \mathcal{A})$, where arcs $a \in \mathcal{A} \subset \{(v, w) \in \mathcal{V}^2 \mid v \neq w\}$ represent pipes and vertices $v \in \mathcal{V}$ represent junctions at which these pipes meet and interact. While we do not assume a fixed direction of flow, the direction of an arc indicates the nominal direction of positive flow in the corresponding pipe. A given pipe

$(v,w) \in \mathcal{A}$ leads from its nominal inlet junction $v \in \mathcal{V}$ to its nominal outlet junction $w \in \mathcal{V}$. We denote the neighborhood of $v \in \mathcal{V}$ by

$$\mathcal{N}(v) := \{w \in \mathcal{V} \mid (v,w) \in \mathcal{A} \vee (w,v) \in \mathcal{A}\}.$$

Junctions also represent locations where the network may exchange gas with the environment. To this end, some junctions are marked as supply junctions $v \in \mathcal{S}$ that inject gas into the network or demand junctions $v \in \mathcal{D}$ that withdraw gas from it. We assume that for supply junctions $v \in \mathcal{S}$, $\left|\mathcal{N}(v)\right| = 1$ holds. To simplify notation of compressor stations, we also prohibit antiparallel arcs, that is,

$$(v,w) \in \mathcal{A} \;\Rightarrow\; (w,v) \notin \mathcal{A} \quad \forall v,w \in \mathcal{V}.$$

2.2 Pipes and Flow Modeling

Single-phase subsonic gas flow is generally modeled by using the Euler equations [4,6,9,11,15,16]. We adopt the following simplified version of the one-dimensional Euler equations, suggested in [11] and adopted for transient stochastic optimal control of gas networks in [15]:

$$\partial_t \begin{pmatrix} p \\ q \end{pmatrix} + \underbrace{\begin{pmatrix} 0 & \frac{c^2}{A} \\ A & 0 \end{pmatrix}}_{=:B} \partial_x \begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} 0 \\ -\alpha \cdot \frac{|q|q}{p} \end{pmatrix}, \tag{2.1}$$

where p denotes the average pressure across the pipe's cross section, q denotes the mass of the gas passing through the pipe's cross section in a given unit of time, A denotes the cross-sectional area of the pipe, and c^2 denotes the square of the speed of sound inside the gas. The coefficient α weighs the friction term, discussed below.

We assume that gas flows at sufficiently low speed and follows the following variant of the ideal gas law:

$$p = z\frac{R}{M}T\rho, \tag{2.2}$$

where z is the gas' compressibility factor, M is its molar mass, T is its temperature, and R is the universal gas constant. We also assume that global bounds exist on both pressure and flow rate, given by $p \in [\check{p}, \hat{p}] \subset \mathbb{R}_{>0}$ and $q \in [-\hat{q}, \hat{q}] \subset \mathbb{R}$, respectively.

Source Term. The source term in (2.1) represents an external force acting on the gas due to friction at the inner wall of the pipe. The coefficient α is given by $\alpha := \lambda c^2/(2AD)$, where D is the pipe's inner diameter, $A = \frac{\pi}{4}D^2$ is its cross-sectional area, and λ is the Darcy friction coefficient, which can be derived from the ratio between D and the pipe wall's effective roughness height ϵ following [14] as $\lambda = \left(2\log\left(\frac{3.7D}{\epsilon}\right)\right)^{-2}$. We assume that gravity does not affect the movement of gas, although this assumption can be relaxed by adding a linear source term depending on the local incline and the pressure of the gas.

Differentiability. To make the solutions of the discretized system twice differentiable with respect to the boundary controls, we make the approximation

$$-\alpha \cdot \frac{|q|q}{p} \approx \varphi_{\alpha,M}(p,q) := -\frac{2\alpha}{\pi}\arctan(Mq)\frac{q^2}{p}.$$

Note that the scaled arctangent deviates from the signum function only for small q, unless p approaches zero.

Boundary Conditions and Existence of Solutions. The existence and uniqueness of continuously differentiable solutions over a limited time horizon in the presence of nonlinear source terms are guaranteed by the results of [7] assuming an appropriate set of boundary conditions. The number and location of the boundary conditions are dictated by the eigenvalues and left eigenvectors of the coefficient matrix B, which has two distinct real eigenvalues $\lambda_{1/2} = \mp c$. The Riemann invariants of the PDE system (2.1) are

$$\xi_1 = \sqrt{\frac{A^2}{c^2+A^2}}p - \sqrt{\frac{c^2}{c^2+A^2}}q \quad \text{and} \quad \xi_2 = \sqrt{\frac{A^2}{c^2+A^2}}p + \sqrt{\frac{c^2}{c^2+A^2}}q. \tag{2.3}$$

Note that this linear transformation of (p, q)-space is bijective with

$$p = \frac{1}{2}\sqrt{\frac{c^2+A^2}{A^2}}(\xi_2 + \xi_1) \quad \text{and} \quad q = \frac{1}{2}\sqrt{\frac{c^2+A^2}{c^2}}(\xi_2 - \xi_1). \tag{2.4}$$

In transformed space, the simplified version of the Euler equations (2.1) with the approximate source term is

$$\partial_t \xi_1 - c \cdot \partial_x \xi_1 = -\sqrt{\frac{c^2}{c^2+A^2}}\varphi_{\alpha,M}\left(\frac{1}{2}\sqrt{\frac{c^2+A^2}{A^2}}(\xi_2+\xi_1), \frac{1}{2}\sqrt{\frac{c^2+A^2}{c^2}}(\xi_2-\xi_1)\right), \tag{2.5a}$$

$$\partial_t \xi_2 + c \cdot \partial_x \xi_2 = \sqrt{\frac{c^2}{c^2+A^2}}\varphi_{\alpha,M}\left(\frac{1}{2}\sqrt{\frac{c^2+A^2}{A^2}}(\xi_2+\xi_1), \frac{1}{2}\sqrt{\frac{c^2+A^2}{c^2}}(\xi_2-\xi_1)\right), \tag{2.5b}$$

meaning that ξ_1 and ξ_2 propagate through the pipe in opposing directions at a fixed speed c. The source term converts ξ_1 into ξ_2 and vice versa, forcing the two toward equilibrium.

It is therefore sufficient to impose a Dirichlet boundary condition on ξ_1 at $x = L$ and a Dirichlet boundary condition on ξ_2 at $x = 0$ [7]. These can be given implicitly by using the boundary pressure on either side of the pipe, which is convenient for modeling compressor stations on the inlet and outlet of each pipe. Together with suitable initial conditions, we thus obtain the following initial and boundary conditions:

$$\begin{aligned} p(0,x) = p_0(x) \text{ and } q(0,x) = q_0(x) \quad & \forall x \in [0, L], \\ p(t,0) = p_{\text{in}}(t) \text{ and } p(t,L) = p_{\text{out}}(t) \quad & \forall t \in [0, T], \end{aligned}$$

where it is assumed that $p_0(0) = p_{\text{in}}(0)$ and $p_0(L) = p_{\text{out}}(0)$. Under these assumptions, it follows from [7] that the system (2.1) has a unique solution within a neighborhood of $t = 0$.

Temporal Discretization. We discretize (2.1) by first transforming it into a system of ordinary differential equations (ODEs) by means of a finite volume method similar to Godunov's scheme [3] and then solving that ODE system by using the explicit Euler method. We distinguish between two discrete time grids: the integer control grid and the sampling grid. The sampling grid is the finer of the two and serves as a time grid for both the discretization of continuous controls and the discretization of state variables in the optimization problem. Depending on the choice of simulation method and the presence of adaptive error control, the simulation grid may actually be finer than the sampling grid. However, this is not reflected in the optimization problem, since we enforce the constraints of our optimization problem on average over each interval of the sampling grid. Thus, state variables can be said to be sampled over intervals of the sampling grid.

Let a finite time horizon $[0, T] \neq \emptyset$ and a number of sampling intervals $N_{\text{t}} \in \mathbb{N}$ be given. For $i \in \{0, \ldots, N_{\text{t}}\}$, the points of the sampling grid are given by $t_i := \frac{iT}{N_{\text{t}}}$ with time step $\Delta t := \frac{T}{N_{\text{t}}}$. Our

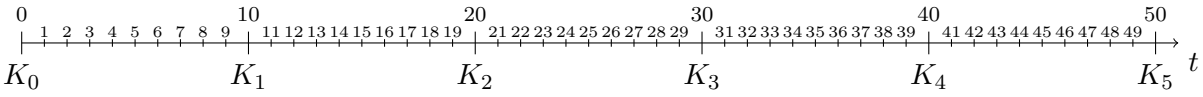


Figure 1: Illustration of sampling and integer control grid ($N_\mathrm{t} = 50$, $\delta = 10$, $N_\mathrm{w} = 5$)

sampling grid is a refinement of the integer control grid, which we also assume to be equidistant; that is, there is a step $\delta \in \{1, \dots N_\mathrm{t}\}$ with $\delta \mid N_\mathrm{t}$ such that the grid points of the integer control grid are given by t_{K_i} with

$$K_i := i\delta \quad \forall i \in \left\{0, \dots, \frac{N_\mathrm{t}}{\delta}\right\}.$$

The relationship between the two time grids is illustrated in Figure 1. For ease of notation, we define $N_\mathrm{w} := \frac{N_\mathrm{t}}{\delta}$. Further, for every sampling interval $j \in \{1, \dots, N_\mathrm{t}\}$ we denote the index of the associated integer control interval by

$$J_j := \left\lfloor \frac{i-1}{\delta} \right\rfloor + 1.$$

Spatial Discretization. Finite volume methods are based on the observation that for continuously differentiable p and q, the following equality holds because of the divergence theorem:

$$\int_{x_1}^{x_2} \left(\partial_t \begin{pmatrix} p \\ q \end{pmatrix} + B \cdot \partial_x \begin{pmatrix} p \\ q \end{pmatrix}\right) \mathrm{d}x = \partial_t \left(\int_{x_1}^{x_2} \begin{pmatrix} p \\ q \end{pmatrix} \mathrm{d}x\right) + B \cdot \begin{pmatrix} p|_{x=x_2} - p|_{x=x_1} \\ q|_{x=x_2} - q|_{x=x_1} \end{pmatrix}. \tag{2.7}$$

We divide each pipe by an equidistant grid, $x_i := i\Delta x := iL/N_\mathrm{x}$ for all $i \in \{0, \dots, N_\mathrm{x}\}$ and denote the averages of p and q over the ith spatial subinterval $[x_{i-1}, x_i]$ by $\bar{p}_i$ and $\bar{q}_i$ for $i \in \{1, \dots, N_\mathrm{x}\}$:

$$\bar{p}_i(t) := \frac{1}{\Delta x} \int_{x_{i-1}}^{x_i} p(t,x)\,\mathrm{d}x, \quad \bar{q}_i(t) := \frac{1}{\Delta x} \int_{x_{i-1}}^{x_i} q(t,x)\,\mathrm{d}x.$$

We replace the pressure and flow rate in the source term by their respective averages and apply the explicit Euler scheme for time integration, thereby deriving the following equations from (2.7):

$$\frac{\bar{p}_{i,j+1} - \bar{p}_{i,j}}{\Delta t} = \frac{c^2}{A} \cdot \frac{q|_{t_j,x_{i-1}} - q|_{t_j,x_i}}{\Delta x} \qquad \forall i \in \{1, \dots, N_\mathrm{x}\}, j \in \{0, \dots, N_\mathrm{t} - 1\}, \tag{2.8a}$$

$$\frac{\bar{q}_{i,j+1} - \bar{q}_{i,j}}{\Delta t} = A \cdot \frac{p|_{t_j,x_{i-1}} - p|_{t_j,x_i}}{\Delta x} + \varphi_{\alpha,M}\left(\bar{p}_{i,j}, \bar{q}_{i,j}\right) \; \forall i \in \{1, \dots, N_\mathrm{x}\}, j \in \{0, \dots, N_\mathrm{t} - 1\}. \tag{2.8b}$$

We then approximate the state function values at the grid points (t_j, x_i). This approximation is made by solving the homogeneous problem locally for small time horizons, which can be done analytically by linearly combining the Riemann invariants from the subintervals immediately incident to the grid point:

$$p|_{t_j,x_i} = \frac{1}{2}\sqrt{\frac{c^2 + A^2}{A^2}}\left(\bar{\xi}_{2,i}|_{t_j} + \bar{\xi}_{1,i+1}|_{t_j}\right) = \frac{1}{2}\left(\bar{p}_{i,j} + \bar{p}_{i+1,j}\right) + \frac{c}{2A}\left(\bar{q}_{i,j} - \bar{q}_{i+1,j}\right),$$

$$q|_{t_j,x_i} = \frac{1}{2}\sqrt{\frac{c^2 + A^2}{c^2}}\left(\bar{\xi}_{2,i}|_{t_j} - \bar{\xi}_{1,i+1}|_{t_j}\right) = \frac{A}{2c}\left(\bar{p}_{i,j} - \bar{p}_{i+1,j}\right) + \frac{1}{2}\left(\bar{q}_{i,j} + \bar{q}_{i+1,j}\right).$$

For these terms to be independent of the states in other subintervals, we have to impose an upper bound on the time step that is equal to the time it takes a pressure wave to cross an entire subinterval. This leads to the following Courant-Friedrichs-Lewy (CFL) condition:

$$\Delta t \leq \frac{\Delta x}{c}. \tag{2.9}$$

For the boundary grid points $i = 0$ and $i = N_{\mathrm{x}}$, we must substitute $\bar{\xi}_{1,N_{\mathrm{x}}+1}|_{t_j}$ and $\bar{\xi}_{2,0}|_{t_j}$ with appropriate expressions derived from the boundary conditions. While boundary pressures are given explicitly, boundary flow rates are determined by linearly combining boundary pressure and the state in the spatial interval directly adjacent to the boundary:

$$\bar{\xi}_{2,0}|_{t_j} := 2\sqrt{\frac{A^2}{c^2+A^2}}p_{\mathrm{in},j+1} - \bar{\xi}_{1,1}|_{t_j}, \quad \text{and} \quad \bar{\xi}_{1,N_{\mathrm{x}}+1} := 2\sqrt{\frac{A^2}{c^2+A^2}}p_{\mathrm{out},j+1} - \bar{\xi}_{2,N_{\mathrm{x}}}|_{t_j},$$

meaning that

$$q|_{t_j,x_0} = \frac{A}{c}\big(p_{\mathrm{in},j+1} - \bar{p}_{1,j}\big) + \bar{q}_{1,j}, \quad \text{and} \quad q|_{t_j,x_{N_{\mathrm{x}}}} = \frac{A}{c}\big(\bar{p}_{N_{\mathrm{x}},j} - p_{\mathrm{out},j+1}\big) + \bar{q}_{N_{\mathrm{x}},j}.$$

By making these substitutions in the discretized Euler equations (2.8), we obtain a suitable system of equality constraints for a given pipe $a \in \mathcal{A}$.

Since the explicit Euler method produces piecewise-linear solutions, the average inlet and outlet flow rates over a time interval specified by the index $j \in \{1, \ldots, N_{\mathrm{t}}\}$ are given by

$$\bar{q}^{(a)}_{\mathrm{in},j} = \frac{A}{c}\Big(p^{(a)}_{\mathrm{in},j} - \frac{1}{2}\big(\bar{p}^{(a)}_{1,j-1} + \bar{p}^{(a)}_{1,j}\big)\Big) + \frac{1}{2}\big(\bar{q}^{(a)}_{1,j-1} + \bar{q}^{(a)}_{1,j}\big), \tag{2.10a}$$

$$\bar{q}^{(a)}_{\mathrm{out},j} = \frac{A}{c}\Big(\frac{1}{2}\big(\bar{p}^{(a)}_{N_{\mathrm{x}},j-1} + \bar{p}^{(a)}_{N_{\mathrm{x}},j}\big) - p^{(a)}_{\mathrm{out},j}\Big) + \frac{1}{2}\big(\bar{q}^{(a)}_{N_{\mathrm{x}},j-1} + \bar{q}^{(a)}_{N_{\mathrm{x}},j}\big). \tag{2.10b}$$

Note that equations (2.8) and (2.10) are completely symmetric with respect to the orientation of the pipe and the designation of inlet and outlet, making them indifferent to flow direction. Since boundary pressures and flows are shared between adjacent intervals, finite volume methods guarantee that, in total, no mass or impulse is lost between cells.

Remark 2.1. *Indifference to direction of flow presents a major advantage of the finite volume method over simple upwinding schemes such as the one used in [15], which can guarantee proper conservation of mass and impulse only if the direction of flow is known ahead of time and may require artificial flow-splitting schemes and the introduction of large numbers of integers in scenarios where the direction of flow is not known ahead of time and may vary across the length of the pipe.*

Initial Conditions and Steady States. For the PDE solution to be well defined, an initial state must be specified. For instance, this state may simply be assumed or derived by interpolating between sparse sensor data. Here, we choose a steady initial state. For a single pipe, this can easily be expressed by demanding that the right-hand side of the ODE system (2.8) be zero at the beginning of the time horizon:

$$0 = \frac{c^2}{A} \cdot \frac{q|_{0,x_{i-1}} - q|_{0,x_i}}{\Delta x} \qquad \forall i \in \{1, \ldots, N_{\mathrm{x}}\}, \tag{2.11a}$$

$$0 = A \cdot \frac{p|_{0,x_{i-1}} - p|_{0,x_i}}{\Delta x} + \varphi_{\alpha,M}\big(\bar{p}_{i,0}, \bar{q}_{i,0}\big) \ \forall i \in \{1, \ldots, N_{\mathrm{x}}\}, \tag{2.11b}$$

where equation (2.11a) implies that the flow rate throughout the pipe is initially constant.

Objective Function Terms. Our goal is that the network bridge the demand fluctuation being modeled without deviating from the stable state that it starts in. Otherwise, the network could avoid drawing in gas from external sources by running a deficit over the time horizon of the simulation. We avoid this situation by introducing an L_1 penalty for net differences in both average pressure and average mass flux. For each pipe $(v,w) \in \mathcal{A}$, we introduce additional variables $\Delta\bar{p}^{(v,w)}, \Delta\bar{q}^{(v,w)} \in \mathbb{R}^{N_\mathrm{x}^{(v,w)}}$ and linear inequality constraints equivalent to

$$\Delta\bar{p}_*^{(v,w)} \geq |\bar{p}_{*,0}^{(v,w)} - \bar{p}_{*,N_\mathrm{t}}^{(v,w)}|, \quad \text{and} \quad \Delta\bar{q}_*^{(v,w)} \geq |\bar{q}_{*,0}^{(v,w)} - \bar{q}_{*,N_\mathrm{t}}^{(v,w)}|. \tag{2.12a}$$

The penalty term for (v,w) then takes the form

$$\kappa\Delta x^{(v,w)} \sum_{i=1}^{N_\mathrm{x}^{(v,w)}} \left(\Delta\bar{p}_i^{(v,w)} + \Delta\bar{q}_i^{(v,w)}\right). \tag{2.13}$$

Note that by fixing the initial state and penalizing the difference to a fixed terminal state, one can use the model to plan any transition between two known network states.

2.3 Junctions and Compressor Stations

Junctions correspond to vertices of the network graph and are therefore points where multiple pipes interact. We assume that junctions themselves do not possess any capacity to store gas, a situation that can be modeled with the following balance equation:

$$\sum_{(w,v)\in\mathcal{A}} q^{(w,v)}(t,L_a) - \sum_{(v,w)\in\mathcal{A}} q^{(v,w)}(t,0) + q_\mathrm{src}^{(v)}(t) - q_\mathrm{snk}^{(v)}(t) = 0 \quad \forall v \in \mathcal{V}, t \in [0,T], \tag{2.14}$$

where $q_\mathrm{src}^{(v)} \colon [0,T] \to [0,\infty)$ denotes the amount of gas injected at junction $v \in \mathcal{V}$ over time and $q_\mathrm{snk}^{(v)} \colon [0,T] \to [0,\infty)$ denotes the amount of gas withdrawn at the same junction over time. Naturally, we assume that source and sink flows are zero at junctions that do not act as sources or sinks.

Junctions also control pipe's boundary pressures through compression. If no compression is performed, all pipes connected to a given junction must share the same boundary pressure. However, we assume that each junction $v \in \mathcal{V}$ also functions as a compressor station that may apply a multiplicative boost to the boundary pressure of each connected pipe:

$$p_\mathrm{bnd}^{(v,w)}(t) = \mu^{(v,w)}(t) p^{(v)}(t) \quad \forall t \in [0,T], (v,w) \in \mathcal{V}^2 \colon a(v,w) \in \mathcal{A}, \tag{2.15}$$

where $a(v,w) = (v,w)$ if $(v,w) \in \mathcal{A}$ and $a(v,w) = (w,v)$ otherwise. Note that $a(v,w) = (v,w)$ exactly if v is the inlet junction of $a(v,w)$. In this case, we define $p_\mathrm{bnd}^{a(v,w)} := p_\mathrm{in}^{(v,w)}$. For the outlet side, we define $p_\mathrm{bnd}^{a(v,w)} := -p_\mathrm{out}^{(w,v)}$. The compression ratio is denoted by $\mu^{(v,w)}(t) \geq 1$.

Accurate modeling of different types of compressor machines exceeds the scope of this paper (see, e.g., [6, Ch. 2]). Note that the compression ratio μ is subject to highly nonlinear and nonconvex constraints that stem from operational constraints of the compressor machinery. We may reasonably assume that the feasible region of each compressor machine is given in terms of the pipe's boundary mass flux and the compressor's discharge pressure, which is equal to the pipe's boundary pressure. Further, we assume that a good polyhedral inner approximation of the feasible region, as depicted in Figure 2, can be found. The feasible region of a compressor machine is

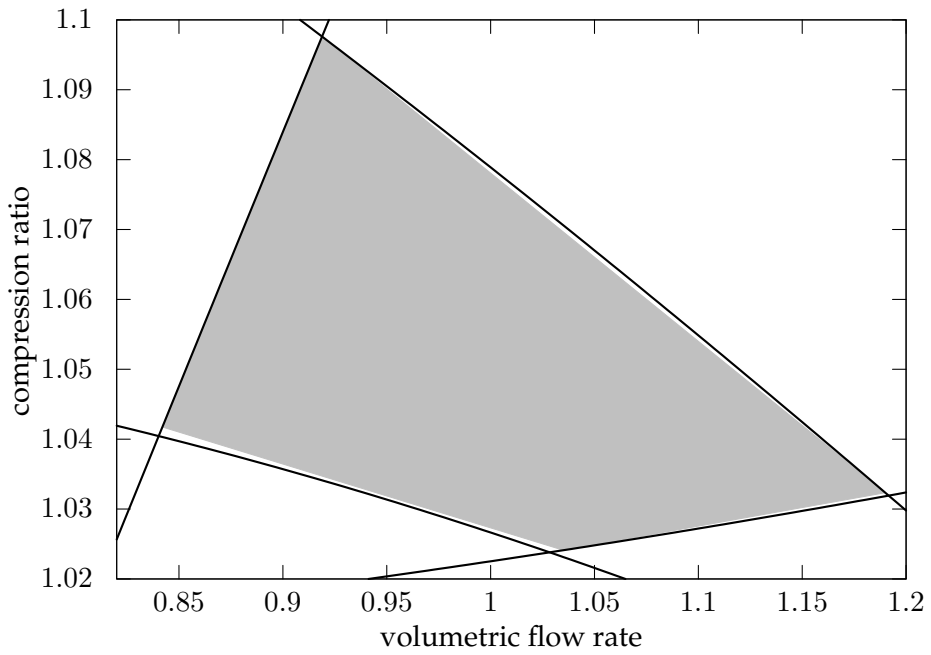


Figure 2: Centrifugal compressor's approximate feasible region (GasLib-40 test network [13]).

therefore approximated by the polyhedral cone given by

$$A \begin{pmatrix} p_{\text{bnd}} \\ q_{\text{bnd}} \end{pmatrix} \leq b p^{(v)}, \text{ or equivalently } \begin{pmatrix} A & -b \end{pmatrix} \begin{pmatrix} p_{\text{bnd}} \\ q_{\text{bnd}} \\ p^{(v)} \end{pmatrix} \leq 0.$$

In larger compressor stations, different types of compressor machines may be used either in parallel or in series, with different arrangements often being used to cover various subsets of the $(p_{\text{bnd}}, q, p^{(v)})$-space. To account for this type of complex behavior, we allow compressors to operate in different modes. For each compressor installed at an endpoint (v, w) of a pipe (v, w) or $(w, v) \in \mathcal{A}$, let $N_{\text{c}}^{(v,w)} \in \mathbb{N}_0$ denote the number of such modes with $N_{\text{c}}^{(v,w)} = 0$, indicating that no compressor is installed. To switch between these modes, we introduce binary control variables.

For each $k \in \{1, \ldots, N_{\text{c}}^{(v,w)}\}$, we introduce a binary indicator variable $\chi_k^{(v,w)} \colon [0, T] \to \{0, 1\}$ that indicates that the kth mode is active at a given point in time. At any given time, at most one node may be active:

$$\sum_{k=1}^{N_{\text{c}}^{(v,w)}} \chi_k^{(v,w)}(t) \leq 1 \; \forall t \in [0, T], (v, w) \in \mathcal{V}^2 \colon a(v, w) \in \mathcal{A}. \tag{2.16}$$

Further, we define a matrix $A^{(v,w,k)} \in \mathbb{R}^{m^{(v,w,k)} \times 3}$ such that the feasible region of the kth mode of the compressor is enforced by the indicator constraint:

$$\chi_k^{(v,w)}(t) = 1 \;\Rightarrow\; A^{(v,w,k)} \begin{pmatrix} p_{\text{bnd}}^{(v,w)}(t) \\ q_{\text{bnd}}^{(v,w)}(t) \\ p^{(v)}(t) \end{pmatrix} \leq 0 \; \forall k \in \{1, \ldots, N_{\text{c}}^{(v,w)}, (v, w) \in \mathcal{V}^2 \colon a(v, w) \in \mathcal{A}\}. \tag{2.17}$$

Note that the sign of the flow rate inverts for the outlet of the pipe, namely, $q_{\mathrm{bnd}}^{(w,v)} = -q_{\mathrm{out}}^{(v,w)}$ for all $(v,w) \in \mathcal{A}$. When no mode is active, we assume that compressors are being bypassed. Therefore, the compression ratio must be 1:

$$\sum_{k=1}^{N_{\mathrm{c}}^{(v,w)}} \chi_k^{(v,w)}(t) = 0 \;\Rightarrow\; \mu^{(v,w)}(t) = 1 \quad \forall t \in [0,T], (v,w) \in \mathcal{V}^2 \colon a(v,w) \in \mathcal{A}. \tag{2.18}$$

Relaxation of the Indicator Constraints. Since indicator constraints are not generally supported by mixed-integer nonlinear programming (MINLP) solvers, we relax them manually using a big-M formulation. In Section 2.2, we had assumed pressure and flow rate to be globally bounded throughout the network, implying $1 \leq \mu \leq \hat{\mu} := \frac{\hat{p}}{\check{p}}$.

Consider any pipe endpoint (v,w) and associated compressor mode $k \in \{1,\dots,N_{\mathrm{c}}^{(v,w)}\}$. For every row $i \in \{1,\dots,m^{(v,w,k)}\}$ of $A^{(v,w,k)}$, we define

$$M_i^{(v,w,k)} := \max \left\{ A_{i,*}^{(v,w,k)} \begin{pmatrix} p_{\mathrm{bnd}}^{(v,w)} \\ q_{\mathrm{bnd}}^{(v,w)} \\ p^{(v)} \end{pmatrix} \;\middle|\; p_{\mathrm{bnd}}^{(v,w)} \in [\check{p},\hat{p}],\ p^{(v)} \in [\check{p},\hat{p}],\ q_{\mathrm{bnd}}^{(v,w)} \in [-\hat{q},\hat{q}],\ p_{\mathrm{bnd}}^{(v,w)} \geq p^{(v)} \right\}.$$

Given the vector $M^{(v,w,k)} \in \mathbb{R}^{m^{(v,w,k)}}$, the indicator constraints (2.17) are equivalent to

$$A^{(v,w,k)} \begin{pmatrix} p_{\mathrm{bnd}}^{(v,w)}(t) \\ q_{\mathrm{bnd}}^{(v,w)}(t) \\ p^{(v)}(t) \end{pmatrix} \leq M^{(v,w,k)} \left(1 - \chi_k^{(v,w)}(t)\right) \;\forall\; t \in [0,T],\ k \in \{1,\dots,N_{\mathrm{c}}^{(v,w)}\},\ (v,w) \in \mathcal{V}^2 \colon a(v,w) \in \mathcal{A}. \tag{2.19}$$

To simplify the reformulation of (2.18) and the formulation of our objective function in Section 2.4, we introduce separate compression ratio variables $\mu_k^{(v,w)} \colon [0,T] \to [1,\infty)$ for each compressor mode k and accumulate them using a sum of logarithms to obtain the following reformulation of (2.15):

$$\log\bigl(p_{\mathrm{bnd}}^{(v,w)}(t)\bigr) = \log\bigl(p^{(v)}(t)\bigr) + \sum_{k=1}^{N_{\mathrm{c}}^{(v,w)}} \log\bigl(\mu_k^{(v,w)}(t)\bigr) \;\forall t \in [0,T], (v,w) \in \mathcal{V}^2 \colon a(v,w) \in \mathcal{A}, \tag{2.20}$$

which, given that all pressures and compression ratios are strictly positive, is equivalent to multiplying them. The indicator constraint (2.18) can then equivalently be enforced by using

$$\mu_k^{(v,w)}(t) \leq 1 + (\hat{\mu} - 1)\chi_k^{(v,w)}(t) \quad \forall t \in [0,T],\ v \in \mathcal{V},\ w \in \mathcal{N}(v),\ k \in \{1,\dots,N_{\mathrm{c}}^{(v,w)}\}. \tag{2.21}$$

Time Discretization. At each junction $v \in \mathcal{V}$, we introduce a discretized version of the flow balance constraint (2.14) for each interval of the sampling grid:

$$\sum_{(w,v)\in\mathcal{A}} \bar{q}_{\mathrm{out},j}^{(w,v)} - \sum_{(v,w)\in\mathcal{A}} \bar{q}_{\mathrm{in},j}^{(v,w)} + q_{\mathrm{src},j}^{(v)} = \bar{q}_{\mathrm{snk},j}^{(v)} \quad \forall j \in \{1,\dots,N_{\mathrm{t}}\}.$$

We relax these constraints by introducing nonnegative slack variables $\bar{q}_{\mathrm{slk},j}^{(v)} \geq 0$, which represent a certain amount of gas to being ejected from the network at junction v:

$$\sum_{(w,v)\in\mathcal{A}} \bar{q}_{\mathrm{out},j}^{(w,v)} - \sum_{(v,w)\in\mathcal{A}} \bar{q}_{\mathrm{in},j}^{(v,w)} + q_{\mathrm{src},j}^{(v)} - \bar{q}_{\mathrm{slk},j}^{(v)} = \bar{q}_{\mathrm{snk},j}^{(v)} \quad \forall j \in \{1,\dots,N_{\mathrm{t}}\},\ v \in \mathcal{V}. \tag{2.22}$$

Ejecting gas from the network expands the feasible domain but will later be penalized by using an objective function term.

Since flow rates are averaged over intervals of the sampling grid, actual flow rates encountered during the simulation may exceed the feasible region of the compressor station. Because of the boundedness of flow rates and pressures, we assume that sufficiently fine sampling grids will prevent such situations from presenting a real risk if sufficient safety margins are maintained while defining the approximate feasible region.

We discretize node pressure $p^{(v)}$ and compression ratio $\mu^{(v,w)}$ as piecewise constant functions on the sampling grid. Similarly, we discretize $\chi^{(v,w)}$ as a piecewise constant vector function on the integer control grid. In other words, for each node $v \in \mathcal{V}$, we obtain a node pressure vector $\bar{p}^{(v)} \in \mathbb{R}^{N_\mathrm{t}}$; and for each pipe endpoint (v,w), we obtain a compression ratio matrix $\mu^{(v,w)} \in \mathbb{R}^{N_\mathrm{c}^{(v,w)} \times N_\mathrm{t}}$ and a compressor mode matrix $\chi^{(v,w)} \in \{0,1\}^{N_\mathrm{c}^{(v,w)} \times N_\mathrm{w}}$.

The discretized compressor constraints are given by the discretized form of the boundary pressure constraints (2.20),

$$\log\bigl(p_{\mathrm{bnd},j}^{(v,w)}\bigr) - \log\bigl(p_j^{(v)}\bigr) - \sum_{k=1}^{N_\mathrm{c}^{(v,w)}} \log\bigl(\mu_{k,j}^{(v,w)}\bigr) = 0 \;\forall j \in \{1,\dots,N_\mathrm{t}\}, (v,w) \in \mathcal{V}^2 \colon a(v,w) \in \mathcal{A}, \tag{2.23a}$$

the constraint (2.21) restricting the compression ratio of inactive compressors,

$$\mu_{k,j}^{(v,w)} - (\hat{\mu} - 1)\chi_{k,j}^{(v,w)} \leq 1 \quad \forall j \in \{1,\dots,N_\mathrm{t}\},\ v \in \mathcal{V},\ w \in \mathcal{N}(v),\ k \in \{1,\dots,N_\mathrm{c}^{(v,w)}\}, \tag{2.23b}$$

the discretization of the constraints (2.19), which define the compressor feasible region,

$$A^{(v,w,k)} \begin{pmatrix} p_{\mathrm{bnd},j}^{(v,w)} \\ \bar{q}_{\mathrm{bnd},j}^{(v,w)} \\ p_j^{(v)} \end{pmatrix} \leq \left(1 - \chi_{k,J_j}^{(v,w)}\right) M^{(v,w,k)} \quad \begin{array}{l} \forall\ t \in \{1,\dots,N_\mathrm{t}\}, k \in \{1,\dots,N_\mathrm{c}^{(v,w)}\}, \\ (v,w) \in \mathcal{V}^2 \colon a(v,w) \in \mathcal{A}, \end{array} \tag{2.23c}$$

and the special ordered set constraint (2.16),

$$\sum_{k=1}^{N_\mathrm{c}^{(v,w)}} \chi_{k,j}^{(v,w)} \leq 1 \quad \forall j \in \{1,\dots,N_\mathrm{w}\},\ v \in \mathcal{V},\ w \in \mathcal{N}(v). \tag{2.23d}$$

Assumptions and Simplifications. To simplify the definition of feasible regions for our test problems in Section 4, we assume that all compressors defined there have exactly one mode (i.e., $N_\mathrm{c}^{(v,w)} = 1$) and that the only constraint on the feasible region is that compressors may not push against the flow:

$$A^{(v,w,1)} := \begin{pmatrix} 0 & 0 & -1 \end{pmatrix}.$$

Under these assumptions, we can simply choose $M^{(v,w,1)} := \hat{q}$. Further, since there are no incoming pipes fixing the node pressure of supply nodes, we introduce for each supply node a single control variable $p_\mathrm{fix}^{(v)} \in [\check{p}, \hat{p}]$ that defines the internal node pressure over the entire time horizon.

Objective Function Terms. We introduce three objective terms for junctions. First, gas supplied by a supply junction $v \in \mathcal{S}$ is associated with a monetary cost proportional to the mass of gas supplied:

$$\Delta t \sum_{j=1}^{N_\mathrm{t}} C_{\mathrm{src},j}^{(v)} \bar{q}_{\mathrm{src},j}^{(v)}, \tag{2.24}$$

where $C_{\mathrm{src}}^{(v)} \in \mathbb{R}_+^{N_\mathrm{t}}$. Note that in the undiscretized problem, this can be interpreted as a linear functional in $L^2([0,T])$.

Second, for every $v \in \mathcal{V}$ and $w \in \mathcal{N}(v)$, the compressor located at the pipe endpoint (v,w) incurs costs associated with the energy used to compress the gas flowing through it. We assume that all compressors are electrically driven and operate at a mode-dependent constant efficiency. We further assume that electrical power costs are the same for all compressors. Under these assumptions, we obtain a nonconvex objective term of the following form [11, 15] for the endpoint (v,w):

$$\Delta t \sum_{j=1}^{N_\mathrm{t} N_\mathrm{c}^{(v,w)}} \sum_{k=1} C_{\mathrm{cmpr},k,j}^{(v,w)} \bar{q}_{\mathrm{bnd},j}^{(v,w)} \left(\left(\mu_{k,j}^{(v,w)} \right)^\omega - 1 \right).$$

Note that this term is zero for $\mu = 1$, which is necessary in order to make sure that inactive compressor modes do not contribute to the objective function. To avoid introducing additional nonconvexities, we assume that the exponent ω, which is always between 0 and 1, is equal to 1. Further, we assume that most boundary flow rates are roughly equal in magnitude and that the adiabatic efficiency for every compressor mode is equal. Under these assumptions, we arrive at a simpler linear objective term:

$$C_{\mathrm{cmpr}} \Delta t \sum_{j=1}^{N_\mathrm{t} N_\mathrm{c}^{(v,w)}} \sum_{k=1} \left(\mu_{k,j}^{(v,w)} - 1 \right), \tag{2.25}$$

where $C_{\mathrm{cmpr}} \geq 0$ is a global parameter.

Third, we penalize the flow balance slack variable in (2.22) for a given junction $v \in \mathcal{V}$ with the following L^1 penalty term:

$$\kappa \Delta t \sum_{j=1}^{N_\mathrm{t}} q_{\mathrm{slk},j}^{(v)}. \tag{2.26}$$

2.4 Optimization Problem

The complete optimization problem is formulated by aggregating the objective terms and constraints described in preceding sections:

$$\begin{aligned}
\underset{\substack{p,\bar{p},\bar{q},p_{\mathrm{in}},p_{\mathrm{out}},p_{\mathrm{fix}},\\ \bar{q}_{\mathrm{in}},\bar{q}_{\mathrm{out}},q_{\mathrm{src}},q_{\mathrm{slk}},\\ \mu,\chi}}{\text{minimize}} \quad & \sum_{v\in\mathcal{S}} (2.24) + \sum_{v\in\mathcal{V}} \Big((2.26) + \sum_{w\in\mathcal{N}(v)} (2.25) \Big) + \sum_{(v,w)\in\mathcal{A}} (2.13) \\
\text{subject to} \quad & \text{PDE constraints (2.8 – 2.10)} && \forall (v,w) \in \mathcal{A}, \\
& \text{Steady-state constraints (2.11)} && \forall (v,w) \in \mathcal{A}, \\
& \text{Absolute value constraints (2.12)} && \forall (v,w) \in \mathcal{A}, \\
& \text{Flow balance constraints (2.22)} && \forall v \in \mathcal{V}, \\
& \text{Compressor constraints (2.23)} && \forall v \in \mathcal{V}, \\
& \check{p} \leq p_j^{(v)} \leq \hat{p} && \forall v \in \mathcal{V}, j \in \{1,\dots,N_\mathrm{t}\}, \\
& \check{p} \leq p_{\mathrm{bnd},j}^{(v,w)} \leq \hat{p} && \forall v \in \mathcal{V}, w \in \mathcal{N}(v), j \in \{1,\dots,N_\mathrm{t}\}, \\
& p_j^{(v)} = p_{\mathrm{fix}}^{(v)} && \forall v \in \mathcal{S}, j \in \{1,\dots,N_\mathrm{t}\}, \\
& 0 \leq q_{\mathrm{src},j}^{(v)} \leq \hat{q} && \forall v \in \mathcal{S}, j \in \{1,\dots,N_\mathrm{t}\}, \\
& 1 \leq \mu_j^{(v,w)} \leq \hat{\mu} && \forall v \in \mathcal{V}, w \in \mathcal{N}(v), j \in \{1,\dots,N_\mathrm{t}\}, \\
& 0 \leq q_{\mathrm{slk},j}^{(v)} && \forall v \in \mathcal{V}, j \in \{1,\dots,N_\mathrm{t}\}.
\end{aligned} \tag{2.27}$$

The optimization problem (2.27) is a discretized mixed-integer PDE-constrained optimization problem, where the integer controls arise as degrees of freedom in the discretization of the time-dependent binary-valued control function χ. Therefore, the problem has features of a mixed-integer optimal control (MIOC) problem. However, efficient approximate solution methods for MIOC problems, such as combinatorial integral approximation [12], generally do not account for constraints such as (2.23) that restrict controls based on the solution of the differential equation. We therefore must use a general-purpose MINLP solver.

3 Customization of the Branch-and-Bound Algorithm

We implement the model described in Section 2 using the AMPL modeling language [2] and solve the resulting problem using the branch-and-bound MINLP solver Minotaur.[1] Although Minotaur is designed to solve convex MINLPs and is not guaranteed to globally solve problems using our gas network model, we believe the application of a theoretically correct optimization method (e.g., spatial branch-and-bound) to be computationally intractible.

The basic branch-and-bound algorithm has been subject to much prior discussion that is too extensive to be repeated here (see, e.g., [1, 8]). Its customization generally focuses on the areas of branching variable selection, tree node selection, and rounding heuristics. To further accelerate Minotaur's branch-and-bound algorithm and reduce tree sizes, we introduce a new node selection strategy that we will refer to as *rebasing laterally adjusted diving* (RLAD).

Several widespread generic node selection strategies exist. The most common are

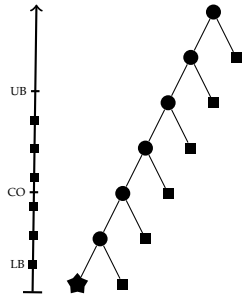
- *depth-first*, which always explores the deepest active node in the tree;
- *breadth-first*, which always explores the most shallow active node in the tree;
- *best-first*, which always explores the active node with the best lower objective bound;
- *best-estimate*, which uses a heuristic objective estimate rather than the lower bound; and
- various types of *diving*, which explore smaller subtrees in a depth-first manner and occasionally backtrack to avoid exhaustive exploration of suboptimal subtrees.

By default, Minotaur uses a diving strategy called *best-then-dive*, which starts a depth-first dive from the active node with the best lower bound and backtracks to the new best active node when encountering any infeasible, fathomed, or integer-feasible nodes. In preliminary experiments, we frequently observed Minotaur abandoning subtrees that could, through manual exploration, be shown to contain nearly optimal solutions, in favor of exploring other subtrees that branch off near the root of the branch-and-bound tree. Since Minotaur defaults to a pseudocost branching strategy, solutions in those subtrees often differed greatly and frequently turned out to be infeasible further down the tree.

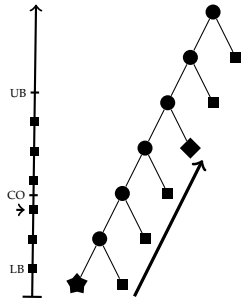
To avoid abandoning promising subtrees, we modify best-then-dive such that, rather than immediately backtracking to the best active node, the solver first attempts to make a small *lateral adjustment* along the dive path to avoid an issue encountered while diving. Taken to the extreme, this approach leads to the depth-first strategy, which can easily degenerate to an exhaustive search. We avoid this by restricting ourselves to lateral adjustments that promise a certain solution quality.

The desired solution quality is expressed as an artifical objective cutoff value f_{co} that lies between the current lower and upper bounds, f_{lb} and f_{ub}, and must be adjusted as those bounds change. We choose $f_{\text{co}} := \min\{\alpha f_{\text{ub}} + (1-\alpha) f_{\text{lb}},\ \beta f_{\text{lb}}\}$ for constants $\alpha \in [0,1]$ and $\beta \geq 1$. However, more sophisticated strategies are conceivable. Node selection using the RLAD strategy is then performed as follows:

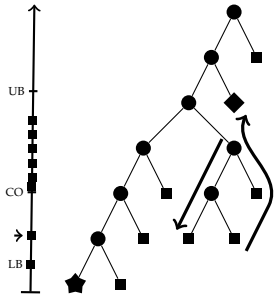
[1] Version 0.2.0, available from https://wiki.mcs.anl.gov/minotaur/ (as of December 18, 2017)



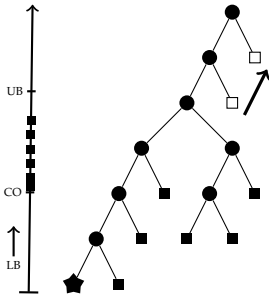
(a) Find first solution using depth-first strategy.



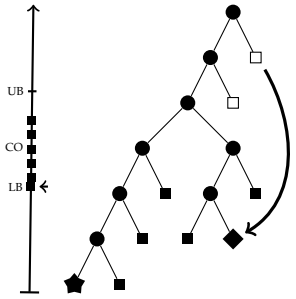
(b) Backtrack gradually, ignoring nodes above the cutoff.



(c) Investigate subtrees rooted in nodes allowed by the cutoff.



(d) Continue until $\mathcal{S}$ is exhausted.



(e) Restart depth-first search at the root of $\mathcal{H}_{\min}$.

Figure 3: Steps of the RLAD tree search (★ integer solution, ◆ next selection, ● non-integer solution, ■ unexplored, □ fathomed).

- Active nodes are stored in a stack $\mathcal{S}$ and a min-heap $\mathcal{H}_{\min}$ sorted by their lower bounds.
- New nodes are always added to the top of $\mathcal{S}$.
- The next node is selected by using Algorithm 1.

During the depth-first segment of our search, we ignore active nodes whose lower bound exceeds f_{co}. We refer to these as *deferred nodes*. When the restricted depth-first search is exhausted, RLAD selects the deferred node with the lowest lower bound. We refer to this step as *rebasing*. Figure 3 illustrates the resulting tree search in a simple example. We note that the introduction of an artificial cutoff is not unique to RLAD but is shared by other *backtracking* node selection strategies, as described in [8]. However, the strategies surveyed in that paper proceed immediately to the rebasing step rather than first attempting to make small adjustments to the dive path.

Algorithm 1: Node Selection in RLAD

```
while S ≠ ∅ do
    Remove n from the top of S;
    if lb(n) ≤ f_co then
        return n;
    else
        Add n to H_min;
Remove the minimal node n from H_min;
return n;
```

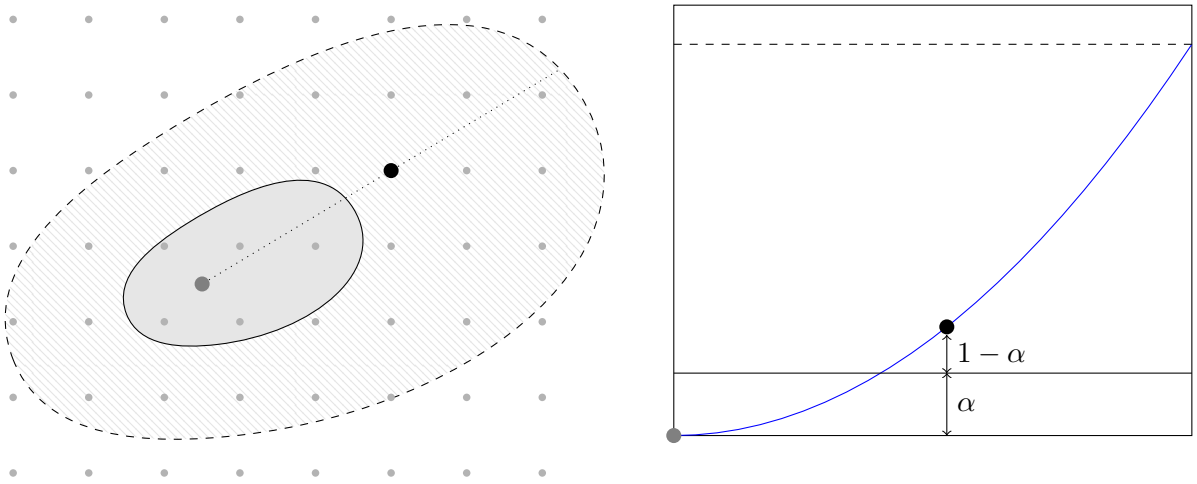


Figure 4: Restriction to decreasing sublevel sets

By enforcing the current cutoff throughout the search, RLAD limits the branch-and-bound search to subtrees with relaxed solutions that lie within a sublevel set of the objective function. The set is decreased as the upper bound decreases and increased as the lower bound increases. In this manner, RLAD performs a full tree search while retaining some of the benefits of a rounding method that attempts to minimize the distance to the solution of the root node relaxation according to a pseudometric induced by the difference in objective function value. Figure 4 illustrates this for a convex objective function.

4 Experiments

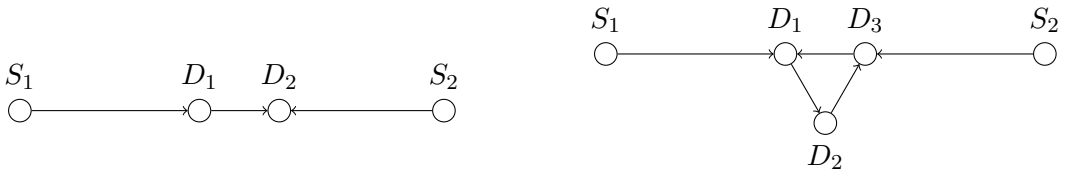
Our model is tested on two test networks, described in Sections 4.1 and 4.2. For each network, we describe a method by which instances can be randomized. Section 4.3 gives parameters that are common to both networks. We then generate six problem instances for each network. Section 4.5 discusses the results of the experiments and their implications for the solution methods proposed in Section 3.

4.1 Test Network "Inversion"

The first test network, "Inversion," is designed to test the model's capability to model the reversal of flow in a pipeline. To the best of our knowledge, deliberate inversion of the direction of flow in a gas network has not been attempted in any other optimal control context. The network consists of three pipelines and four junctions, as depicted in Figure 5a. Pipe-specific parameters are given in Table 1a. The time horizon of the problem is fixed to one hour.

Nodes S_1 and S_2 act as supply nodes. Both can supply between $\check{q}_{\text{src}}^{(S_1)}(t) = \check{q}_{\text{src}}^{(S_2)}(t) = 0.0\,\text{SCM/d}$ and $\hat{q}_{\text{src}}^{(S_1)}(t) = \hat{q}_{\text{src}}^{(S_2)}(t) = 4.5 \times 10^6\,\text{SCM/d}$ of natural gas to the network. Nodes D_1 and D_2 act as demand nodes. Their individual demands are randomized according to a normal distribution with a standard deviation $\sigma = 0.25 \times 10^6\,\text{SCM/d}$ and a mean that changes over time as follows.

- For the first 15 minutes, the mean demand for both D_1 and D_2 is $4.0 \times 10^6\,\text{SCM/d}$.



(a) Test network "Inversion." (b) Test network "Triangle."

Figure 5: Network graphs for both test networks.

Table 1: Pipe parameters for both test networks.

(a) Pipe parameters for "Inversion" network.

Pipe	L_a [km]	D_a [mm]	ϵ_a [mm]	$N_{\mathrm{x}}^{(a)}$
S_1D_1	300	920	0.025	10
D_1D_2	100	920	0.025	6
S_2D_2	300	920	0.025	10

(b) Pipe parameters for "Triangle" network.

Pipe	L_a [km]	D_a [mm]	ϵ_a [mm]	$N_{\mathrm{x}}^{(a)}$
S_1D_1	300	920	0.025	10
D_1D_2	100	920	0.025	6
D_2D_3	100	920	0.025	6
D_3D_1	100	920	0.025	6
S_2D_3	300	920	0.025	10

- From minute 16 to minute 30, the mean demand for D_1 is $6.0 \times 10^6\,\mathrm{SCM/d}$, while the mean demand for D_2 is $2.0 \times 10^6\,\mathrm{SCM/d}$.
- From minute 31 to minute 45, the mean demand for D_1 is $2.0 \times 10^6\,\mathrm{SCM/d}$, while the mean demand for D_2 is $6.0 \times 10^6\,\mathrm{SCM/d}$.
- From minute 46 to minute 60, the mean demand for both D_1 and D_2 is $4.0 \times 10^6\,\mathrm{SCM/d}$.

Any sample drawn from these distributions that deviates from the mean by more than 2σ is redrawn. This strategy guarantees that the sum of both demands can always be met by the maximal total supply. The demand pattern is shown in Figure 6.

Since a single demand of between $5.5 \times 10^6\,\mathrm{SCM/d}$ and $6.5 \times 10^6\,\mathrm{SCM/d}$ cannot be met by a single supply node, part of the higher of the two demands between minute 16 and minute 45 must be met by gas traveling through the pipe connecting D_1 and D_2. By switching the roles of D_1 and D_2, we ensure that a flow reversal in that pipe is required to meet demand.

Problem instances using the "Inversion" network are designated `inversion-base` for an instance without any random fluctuation or `inversion-random`i for $i \in \{\texttt{01}, \texttt{02}, \texttt{03}, \texttt{04}, \texttt{05}\}$.

4.2 Test Network "Triangle"

The second test network, "Triangle," is designed to test the behavior of our model for networks that contain cycles but do not strictly require flow inversions. It consists of seven pipelines and five junctions, as depticted in Figure 5b. Parameters for individual pipes are given in Table 1b. Again, nodes designated S_i for $i \in \{1, 2\}$ act as supply nodes, while nodes designated D_i for $i \in \{1, 2, 3\}$ act as demand nodes. Both demand flow rates and supply costs are subject to normally distributed random fluctuation where the mean value is subject to a sinusoidal fluctuation with a period of one hour, which is equal to the length of the time horizon.

The cost of gas at supply nodes S_1 and S_2 is given by

$$C_{\mathrm{src}}^{(S_i)}(t) := 0.265\,\frac{\mathrm{USD}}{\mathrm{SCM}} + 0.053\,\frac{\mathrm{USD}}{\mathrm{SCM}} \cdot \sin\left(\phi_i + t \cdot 2\pi\,\mathrm{h}^{-1}\right) + X_i,$$

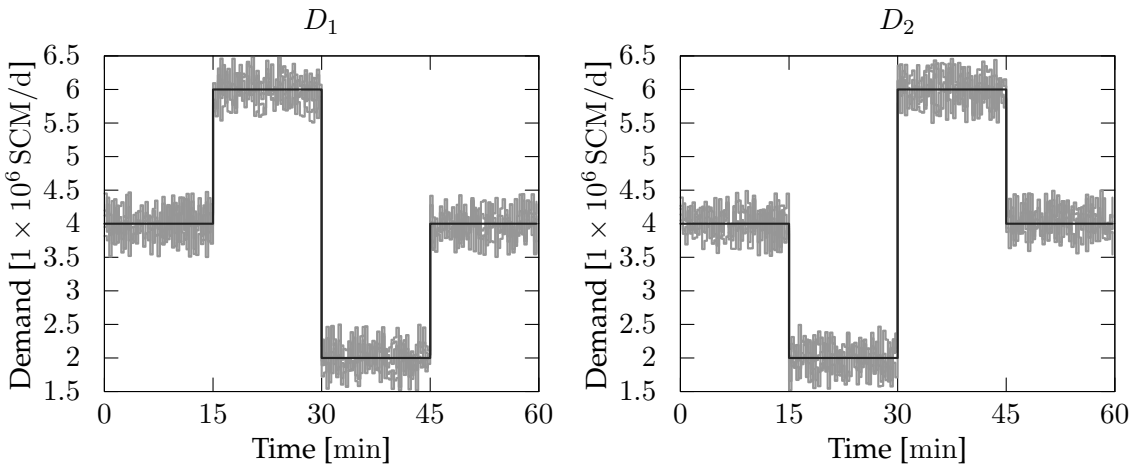


Figure 6: Demand patterns for "Inversion" instances. Black line indicates mean.

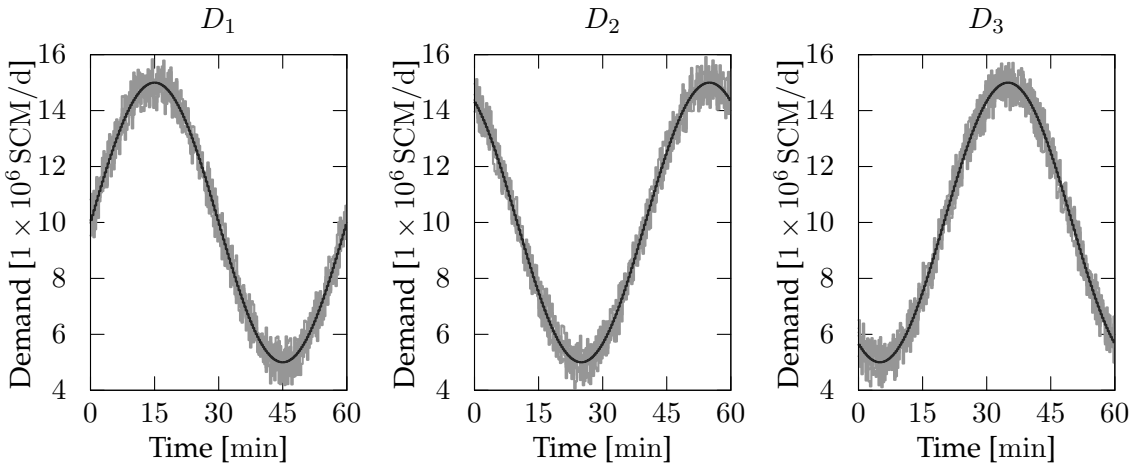


Figure 7: Demand patterns for "Triangle" instances. Black line indicates mean.

where X_i denotes a random variable that is drawn from a normal distribution with a mean of $0.0\,\mathrm{USD/SCM}$ and a standard deviation of $0.018\,\mathrm{USD/SCM}$. If the resulting cost is below $0.071\,\mathrm{USD/SCM}$, the random variable is redrawn. The phase angles are $\phi_1 = 0$ and $\phi_2 = \pi/2$.

The demand at a given demand node D_i with $i \in \{1, 2, 3\}$ is given by

$$q_{\mathrm{snk}}^{(D_i)}(t) := 10 \times 10^6 \,\frac{\mathrm{SCM}}{\mathrm{d}} + 5 \times 10^6 \,\frac{\mathrm{SCM}}{\mathrm{d}} \cdot \sin\left(\psi_i + t \cdot 2\pi\,\mathrm{h}^{-1}\right) + Y_i,$$

where Y_i denotes a random variable that is drawn from a normal distribution with a mean of $0.0\,\mathrm{SCM/d}$ and a standard deviation of $0.5 \times 10^6\,\mathrm{SCM/d}$, which is redrawn if it deviates from the mean by more than two standard deviations. The phase angles are $\psi_i = \frac{2\pi}{3}(i-1)$, ensuring that the sum of the mean demands in all three demand nodes is always $30 \times 10^6\,\mathrm{SCM/d}$. The maximum supply flow from either supply node is $24 \times 10^6\,\mathrm{SCM/d}$, ensuring that the maximum momentary demand of $48 \times 10^6\,\mathrm{SCM/d}$ can be met. This demand pattern is shown in Figure 7.

Problem instances using the "Triangle" network are designated `triangle-base` for an instance without any random fluctuation or `triangle-random`i for $i \in \{\texttt{01}, \texttt{02}, \texttt{03}, \texttt{04}, \texttt{05}\}$.

Table 2: Numbers of variables (total and integer), constraints (equality, inequality, linear, nonlinear), and nonzeros (constraint Jacobian, objective gradient) for the two test problems.

Problem	Var	Int	Eq	Ineq	Lin	Nonlin	NZ in Constr	NZ in Obj
Inversion	13,486	60	11,572	2,264	8,050	5,786	79,432	2,212
Triangle	19,694	100	16,816	3,752	11,890	8,678	118,764	3,136

4.3 Common Parameters

Both test networks are optimized on a time horizon of one hour, which is partitioned into 180 equally sized intervals for the purposes of both simulation and sampling. Therefore, every simulation and sampling interval corresponds to a timespan of 20 seconds. In order to keep the number of integer control variables small, the number of integer control intervals is limited to 10, such that integer controls remain constant over timespans of 6 minutes.

The natural gas flowing in the network is assumed to be pure methane, which has the following physical parameters, largely adopted from [15]:

$$z = 0.80, T = 293.15\,\mathrm{K}, M = 18.0\,\frac{\mathrm{kg}}{\mathrm{kmol}}, \rho_0 = 0.72\,\frac{\mathrm{kg}}{\mathrm{SCM}}.$$

The conversion factor ρ_0 corresponds to the density of methane under standard conditions. Note that along with a universal gas constant of $R = 8{,}314.4598\,\mathrm{J/(kmol\,K)}$, this yields a speed of sound of $c = \sqrt{\frac{zRT}{M}} = 329.13\,\mathrm{m/s}$, meaning that the CFL condition (2.9) is equivalent to $\Delta x \geq 6.583\,\mathrm{km}$, which is satisfied by the pipe parameters laid out in Table 1.

4.4 Experimental Setup

All test instances are solved by using a modified version of Minotaur 0.2.0, using IPOPT 3.12 as a relaxation solver. The exact dimensions of the resulting optimization problems are given in Table 2. Linear equation systems within IPOPT are solved by using the direct solver MA57 from the Harwell Subroutine Library,[2] using a single-threading implementation of BLAS and LAPACK provided by the COIN-OR[3] project. Therefore, all optimization solvers are entirely single-threading with no attempt of parallelization.

Test instances are solved on single compute nodes with Intel E5430 Xeon CPUs and 32 GB of DDR2 667 MHz (4 modules of 8 GB) memory each. Our primary performance metric is wall time, which may be unreliable because the compute nodes are shared with other applications.

We employ six methods of solving the problem:

- `bthend-fwdbr`: Best-then-dive tree search with forward-in-time branching order;
- `bthend-revbr`: Best-then-dive tree search with reverse-in-time branching order;
- `bthend-stdbr`: Best-then-dive tree search with normal reliability branching;
- `rlad-fwdbr`: RLAD tree search with forward-in-time branching order;
- `rlad-revbr`: RLAD tree search with reverse-in-time branching order; and
- `rlad-stdbr`: RLAD tree search with normal reliability branching.

[2] Version 2014.01.10, available at `http://www.hsl.rl.ac.uk/`

[3] See `https://www.coin-or.org/`

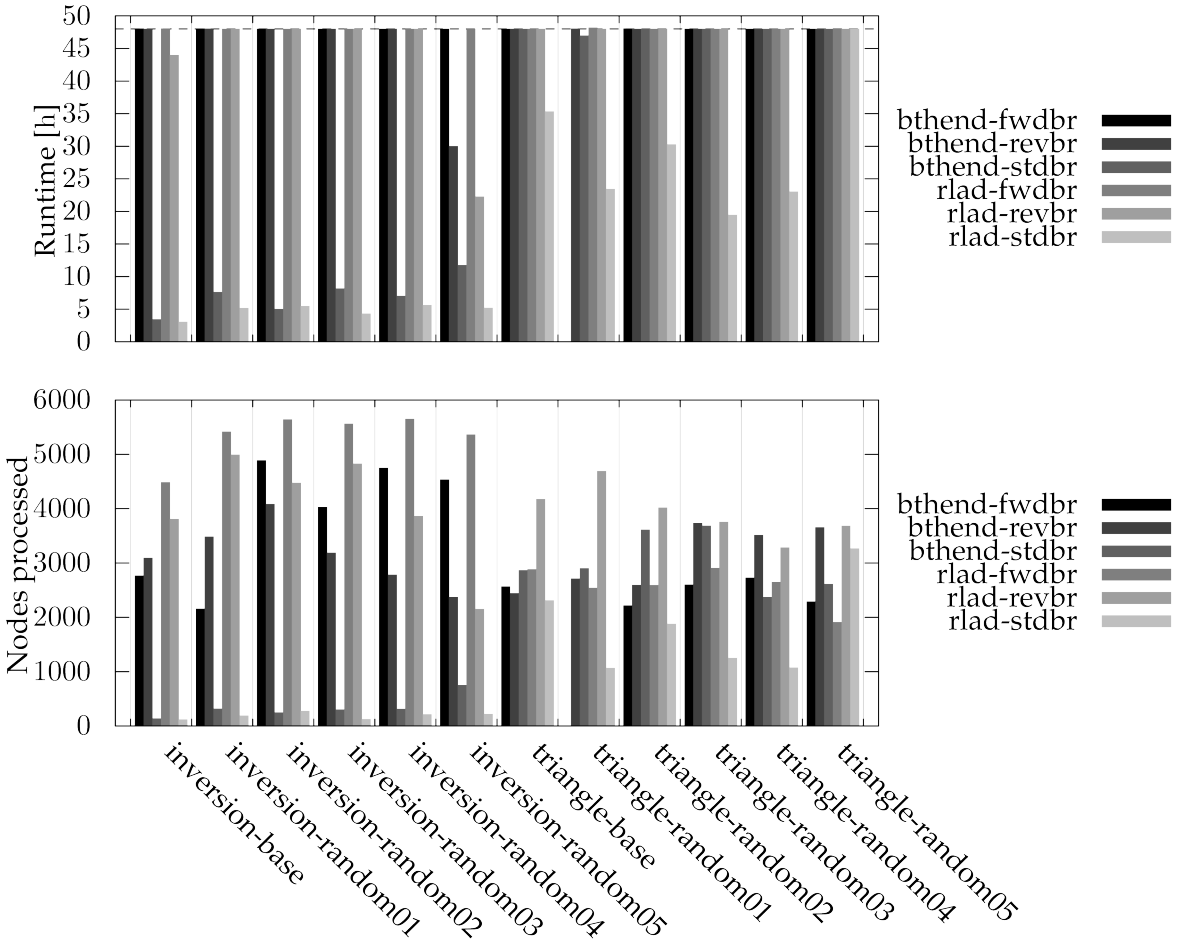


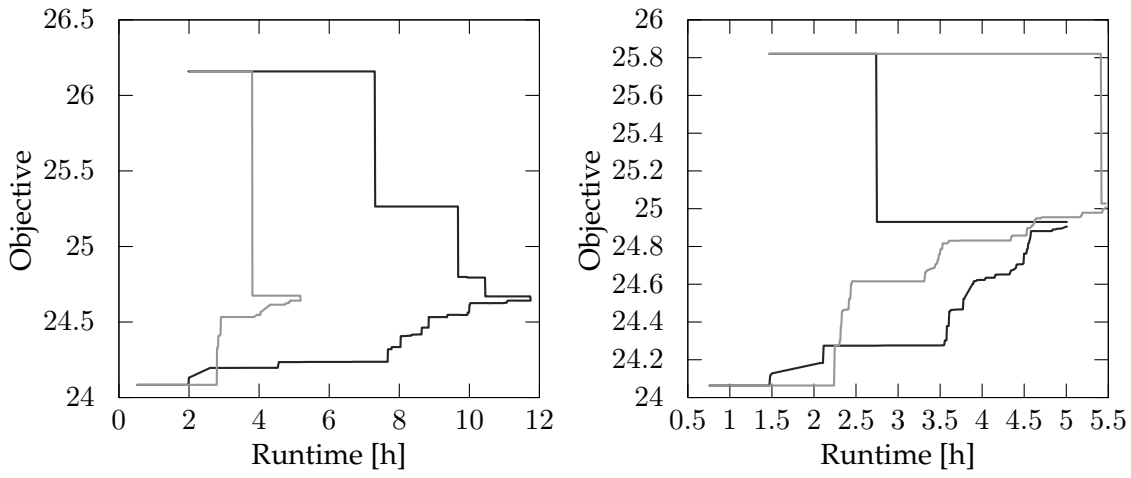
Figure 8: Runtime and number of nodes processed (dashed line indicates runtime limit).

For RLAD, we use an initial target gap $\beta = 1.0$ and a target gap decay rate of $\alpha = 0.1$. The branch-and-bound algorithm is prematurely terminated if the relative integer gap falls below $0.1\,\%$ or the tree search lasts for more than 48 hours.

4.5 Experimental Results

Figure 8 shows the total runtime and number of branch-and-bound nodes processed for each instance and solution method. We omit the results for best-then-dive with forward-in-time branching on instance `triangle-random02` because IPOPT exceeds the maximum number of iterations on most subproblems for this method, thus leading to a misleadingly low number of processed nodes. Most notable in the remaining results is the fact that except for `inversion-random05`, no branching rule other than conventional reliability branching terminates in under 48 hours for any instance or solution method, indicating that both forward- and reverse-in-time branching do not significantly improve the overall performance of the solver. The fact that this is the case for both best-then-dive and RLAD appears to indicate that this is not due to a negative interaction between RLAD and time-based branching orders.

In cases where standard reliability branching is used, RLAD appears to provide a significant advantage in both total runtime and the number of nodes processed to reach a solution. It per-



(a) `inversion-random05` (b) `inversion-random02`

Figure 9: Progression of objective bounds over time for best-then-dive (black) and RLAD (gray) with standard branching order.

forms better than best-then-dive in almost all instances. For `inversion-random05`, the progress of upper and lower bounds over time can be seen in Figure 9a. The instance `inversion-random02` is a notable exception to this trend. Figure 9b shows the progression of global objective bounds over time for this particular instance. It appears that, likely because of nonconvexity, best-then-dive and RLAD find different solutions in this instance. This does not occur in any other test instance.

Flow Reversals We now turn our attention to the solution generated for `inversion-base` using `rlad-stdbr`, which serves as an example of the "Inversion" problem. As noted in Section 4.1, this network was specifically designed to test our model's ability to find solutions that include flow inversions, which are generally hard to achieve because of the operational constraints of compressor machines. Since most compressor machines cannot push directly against the flow of gas, more complex interaction with other network components is needed in order to achieve a flow inversion. This can be observed in the solution.

Figure 10 show the pipe boundary pressures at the inlet sides of the supply pipes (S_1, D_1) and (S_2, D_2). Given a speed of sound of $c = 329.13\,\mathrm{m/s}$ and a pipe length of $L = 300\,\mathrm{km}$ for both pipes, the expected travel time for a pressure wave is between $900\,\mathrm{s}$ and $920\,\mathrm{s}$, or slightly over $15\,\mathrm{min}$. By shifting the inlet pressure profile by this interval, we approximate the projected outlet pressure assuming constant initial pressure and complete absence of friction. We see that inlet pressure at S_2 spikes immediately after the beginning of the simulation and around $30\,\mathrm{min}$ later. The first pressure wave is projected to reach D_2 approximately $15\,\mathrm{min}$ after the beginning of the simulation, causing flow in the exchange pipe (D_1, D_2) to become negative. Similarly, the second spike reaches D_2 at around $45\,\mathrm{min}$, at which point it slows the flow in the exchange pipe and returns the network to a steady state. S_1 emits a pressure spike at around $15\,\mathrm{min}$ that would reach D_1 at around $30\,\mathrm{min}$, when the flow in the exchange pipe reverses direction.

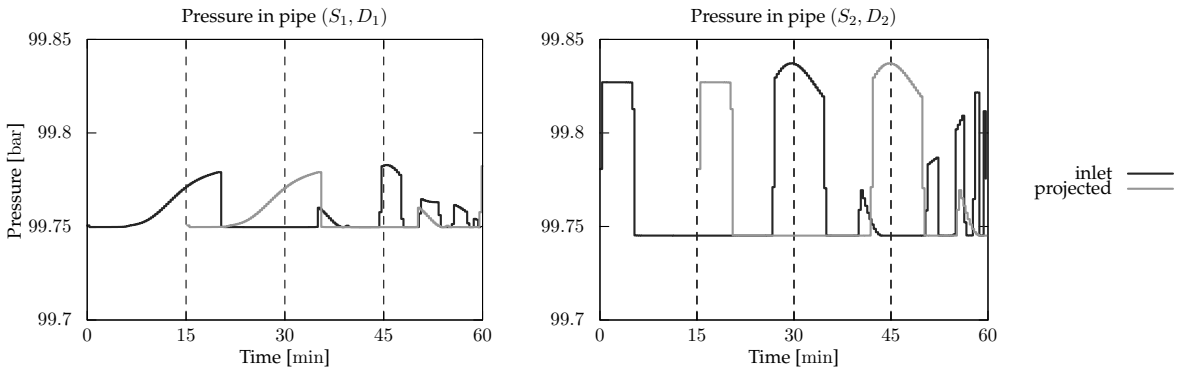


Figure 10: Inlet and projected outlet pressures at (S_1, D_1) and (S_2, D_2).

5 Conclusions and Outlook

In this paper, we develop and demonstrate an optimal control model for gas transport networks. Our overall goal is to find inexpensive control configurations that satisfy fluctuating demand. We model the flow of gas through these networks using a system of partial differential equations derived from the Euler equations, which we discretize using a finite volume method, making our model independent of prior knowledge of direction of flow, thus broadening the range of allowed inputs beyond what is allowed by comparable models. By using our model, we can gain interesting insights into ways to produce more severe changes in the network state, such as flow reversals.

To the best of our knowledge, this is the first model that allows the simulation of gas networks that require flow reversal, which is challenging to achieve. We demonstrate this capability in a small network.

In modeling the compressor stations used to steer the flow of gas in our network, we introduce time-dependent binary switches to the problem, thus making our problem part of the challenging field of mixed-integer PDE-constrained optimization. While mixed-integer optimal control problems can often be solved by using good approximate solution methods, our problem is inaccessible to these methods because of the complex relationship between controls and PDE solutions. We introduce several modifications to branch-and-bound solvers that exploit the same structural aspects as approximate solution methods to accelerate the solution process.

In several randomized instances of two test problems, we show that, particularly by applying our modified tree search method, we can substantially improve the performance of branch-and-bound solvers. However, this tree search method appears to negatively interact with branching orders that attempt to take into account the direction of time, which is commonly exploited in approximate solvers for mixed-integer optimal control.

Future avenues of research include ways of dynamically choosing the cutoff point for gradual backtracking to adapt to small integer optimality gaps. To scale the model to larger networks, we plan to develop implementations using dedicated simulators and parallelization using distributed scientific computing libraries, specifically the PETSc toolkit.

Acknowledgments

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, under Contract DE-AC02-06CH11357. This

work was also supported by the U.S. Department of Energy through grant DE-FG02-05ER25694.

References

[1] Pietro Belotti, Christian Kirches, Sven Leyffer, Jeff Linderoth, James Luedtke, and Ashutosh Mahajan. Mixed-integer nonlinear optimization. *Acta Numerica*, 22:1–131, 2013.

[2] Robert Fourer, David M Gay, and Brian W Kernighan. *AMPL: A mathematical programming language*. AT&T Bell Laboratories, Murray Hill, NJ 07974, 1987.

[3] Sergei Konstantinovich Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 89(3):271–306, 1959.

[4] Martin Gugat, Günter Leugering, Alexander Martin, Martin Schmidt, Mathias Sirvent, and David Wintergerst. MIP-based instantaneous control of mixed-integer PDE-constrained gas transport problems. *Preprint*, 2017.

[5] Benjamin Hiller, René Saitenmacher, and Tom Walther. Analysis of operating modes of complex compressor stations. In *Operations Research Proceedings 2016*, pages 251–257. Springer, 2018.

[6] Thorsten Koch, Benjamin Hiller, Marc E Pfetsch, and Lars Schewe. *Evaluating gas network capacities*. SIAM, 2015.

[7] Ta-Tsien Li and Yi Jin. Semi-global C1 solution to the mixed initial-boundary value problem for quasilinear hyperbolic systems. *Chinese Annals of Mathematics*, 22(03):325–336, 2001.

[8] Jeff T Linderoth and Martin WP Savelsbergh. A computational study of search strategies for mixed integer programming. *INFORMS Journal on Computing*, 11(2):173–187, 1999.

[9] Alexander Martin, Markus Möller, and Susanne Moritz. Mixed integer models for the stationary case of gas network optimization. *Mathematical Programming*, 105(2-3):563–582, 2006.

[10] A Osiadacz. Nonlinear programming applied to the optimum control of a gas compressor station. *International Journal for Numerical Methods in Engineering*, 15(9):1287–1301, 1980.

[11] Andrzej Osiadacz. Simulation of transient gas flows in networks. *International Journal for Numerical Methods in Fluids*, 4(1):13–24, 1984.

[12] Sebastian Sager, Michael Jung, and Christian Kirches. Combinatorial integral approximation. *Mathematical Methods of Operations Research*, 73(3):363, 2011.

[13] Martin Schmidt, Denis Aßmann, Robert Burlacu, Jesco Humpola, Imke Joormann, Nikolaos Kanelakis, Thorsten Koch, Djamal Oucherif, Marc E. Pfetsch, Lars Schewe, Robert Schwarz, and Mathias Sirvent. Gaslib – a library of gas network instances. *Data*, 2(4), 2017.

[14] PK Swanee and Akalank K Jain. Explicit equations for pipeflow problems. *Journal of the Hydraulics Division*, 102(5):657–664, 1976.

[15] Victor M Zavala. Stochastic optimal control model for natural gas networks. *Computers & Chemical Engineering*, 64:103–113, 2014.

[16] Anatoly Zlotnik, Michael Chertkov, and Scott Backhaus. Optimal control of transient flow in natural gas networks. In *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*, pages 4563–4570. IEEE, 2015.